REMARKS

The Office Action dated January 20, 2006 has been received and carefully noted. The above amendments to the claims, and the following remarks, are submitted as a full and complete response thereto.

Claims 1, 6 and 10 have been amended to more particularly point out and distinctly claim the invention. No new matter has been added, and no new issues are raised which require further consideration and/or search. Claims 1-13 are submitted for consideration.

Claims 1-13 were rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. Specifically, the Office Action alleges that claims 1, 6 and 10 recites the limitation of "an insertion module configured to insert null bytes into the header of the cell of the data packet to form a modified header cell of the data packet if the counter determines that the cell of the data packet does not satisfy the multiple of the predetermined number of bytes." According to the Office Action, the an insertion module inserts null bytes into the cell of the data packet to form a modified cell of the data packet and not a modified header cell as recited in the claims. The last two sentences of paragraph 0041 of the specification states that "to prevent data misalignment from occurring, device 105 may insert eight null bytes into the header cell to replace the eight byte header that was removed. As a result of this null-byte insertion technique, this modified header cell will cause the remaining incoming cells to be aligned

to the burst size mandated by the aggregator." Claims 1, 6 and 10 have been amended to overcome this rejection. Therefore, Applicants request that this rejection be withdrawn.

Claims 1, 6 and 10 were rejected under 35 U.S.C. 103(a) as being obvious over European Patent No. 0 572 145 A2 to Thompson in view of U.S. Patent No. 6,512,773 B1 to Scott. According to the Office Action, Thompson teaches all of the elements of claims 1, 6 and 10 except for a counter to determine whether the cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed. Thus, the Office Action uses Scott to cure these deficiencies of Thompson and combines the teachings of Scott and Thompson to yield the elements of claims 1, 6 and 10. The rejection is traversed as being based on references that neither teach nor suggest the novel combination of features clearly recited in independent claims 1, 6 and 10.

Claim 1, upon which claims 2-5 depend, recites a network device that is configured to prevent data misalignment of a data packet containing extra header bytes. The network device includes an ingress module having an input interface to receive a cell of the data packet and a header detector configured to detect a header cell of the data packet and remove a header from the header cell of the data packet. The network device also includes a counter to determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed. The network device further includes an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the cell of the data packet does not satisfy the multiple of the

predetermined number of bytes. The network device also includes an extraction module configured to remove the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device.

Claim 6, upon which claims 7-9 depend, recites a method of preventing data misalignment of a data packet containing extra header bytes. The method includes receiving a cell of the data packet at an input port of a network device and detecting a header cell of the data packet. The method also includes removing a header from the header cell of the data packet and determining whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed. The method further includes inserting null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the cell of the data packet does not satisfy the multiple of the predetermined number of bytes and forwarding the modified cell of the data packet to an output port. The method also includes removing the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device.

Claim 10, upon which claims 11-13 depend, recites a network device configured to prevent data misalignment of a data packet containing extra header bytes. The network device includes receiving means for receiving a cell of the data packet at an input port of the network device and detecting means for detecting a header cell of the data packet. The network device also includes header removing means for removing a header from the header cell of the data packet and determining means for determining

whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed. The network device further includes inserting means for inserting null bytes into the header cell packet to form a modified header cell of the data packet if the counter determines that the cell of the data packet does not satisfy the multiple of the predetermined number of bytes and forwarding means for forwarding the modified cell of the data packet to an output port. The network device also include null byte removing means for removing the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device.

As will be discussed below, the cited prior art references of Thompson and Scott fail to disclose or suggest the elements of any of the presently pending claims.

As discussed in Applicants' previous Response, Thompson teaches a computer system with a processor, a cache, a memory and a network adapter. The network adapter generates and inserts network data checksums. In the outbound direction, the processor provides checksum control information to the network adapter and the network adapter calculates the checksum and inserts the checksum into the proper location within the packet before transmitting the packet on the network. In the inbound direction, the network adapter decodes the packet header, programs the checksum control information directly into internal registers, calculates the checksum and inserts the checksum into the proper location within the packet before transmitting the packet on the memory. The network adapter also automatically separates headers and data during transfer of

incoming packets from the adapter to the memory. The network data further performs alignment of network headers by inserting pad bytes based on specific values found in the network link header. Col. 3, line 1-Col 4, line 50.

The network adapter is connected to the network through a front plane controller that provides transmission and reception of data packets to and from the network. For outbound transfers, the front plane controller unpacks the words from a DMA bus, looks at the first byte of the output stream, which contains a count of how many pad bytes were inserted in the packet and strips off the pad bytes. Col. 6, lines 35-46.

As discussed in Applicants' previous Response, Scott teaches an improved system and method for transporting information over a communication channel. Scott uses a first frame 100 which includes a payload that includes user data PDU to which is prepended a 4-octet ATM header that indicates that the frame is a low overhead cell frame. A trailer is also appended to the frame. Col. 8, lines 18-37. Scott also uses a second frame 150 which includes one of a plurality of 52-octet ATM cells to which is added a header, which indicates that the payload is framed cells, and a trailer. Col. 9, lines 10-27. The system includes a central transceiver which receives either frame 100 or 150 from a remote transceiver over a subscriber line. Figure 5C illustrates the steps performed at the central processor to implement a SAR (segmentation and reassembly) process. First, the payload is processed from frame 100 (block 231) and the number of octets of the user data PDU of the payload is counted (block 232). A user-to-user field and a common part indicator field are formed for the AAL5 frame (block 234). If the

user-to user field and the common part indicator field are not included in the header or trailer, the default "0" is used. Pad characters are added to make the AAL5 frame equal an integer number of 48 octet cells (block 236). The 32-bit cyclic redundancy check of the AAL5 frame is calculated (block 237) and the AAL5 frame is segmented into an integer number of 48 octet cells (block 238). Thereafter, the ATM header from the payload is extracted (block 239). A HEC is added to the 4 octet ATM header to form a 5 octet ATM header (block 241) which is prepended to the 48 octet cells (block 242). Col. 10, lines 16-58.

Applicants submit that the combination of Thompson and Scott simply does not teach or suggest the combination of features clearly recited in claims 1-13. Each of claims 1, 6, and 10, in part, recites a header detector configured to detect a header cell of the data packet and remove a header from the header cell of the data packet; a counter to determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed; an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes. The Office Action admits that Thompson does not teach at least the element of determining whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed, as recited in claims 1, 6, and 10. However, the Office Action cites Col. 10, lines 40-50 and Figure 5C, item #236 of Scott as teaching

determining whether the cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed as recited in claims 1, 6, and 10. As noted above, Col. 8, lines 18-37 of Scott uses a first frame 100 which includes a payload that includes user data PDU to which is prepended a 4-octet ATM header that indicates that the frame is a low overhead cell frame. A trailer is also appended to the frame. Col. 9, lines 10-27 of Scott also uses a second frame 150 which includes one of a plurality of 52-octet ATM cells to which is added a header, which indicates that the payload is framed cells, and a trailer. There is no teaching or suggestion in Scott of a packet that is divided into cells, wherein the header cell of the packet includes a header and packet data information and the rest of the packet data information is included in other cells, as disclosed in the present invention. As disclosed on page 21, paragraph 0040 of the present invention, data packet 106 is programmed to be one hundred and twenty-eight-bytes. Paragraph 0041 of the present invention further discloses that the aggregator is optimized to accommodate, for example sixth-four-burst size. Thus, the one hundred and twenty eight byte packet is divided into two cells, the header cell including a header and packet data and the other including the remaining packet data. As mentioned above, there is simply no teaching or suggestion in Scott of dividing a packet into multiple cells, including a header cell, as disclosed in the present invention. Thus, there is also no teaching or suggestion in Scott of processing a header cell. Specifically, there is no teaching or suggestion in Scott of a header detector configured to detect a header cell of the data packet and remove a header from the header cell of the data packet; a counter to

determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed; an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes, as recited in claims 1, 6 and 10. Therefore, Applicant respectfully asserts that the rejection under 35 U.S.C. §103(a) should be withdrawn because neither Thompson nor Scott, whether taken singly or combined, teaches or suggests each feature of claims 1, 6, 10.

Claims 2-3, 7-8 and 11-12 were rejected under 35 U.S.C. 103(a) as being obvious over Thompson in view of Scott and further in view of U.S. Patent No. 6,567,413 B1 to Denton. According to the Office Action, Thompson and Scott teach all of the elements of claims 1, 6 and 10 except for teaching an aggregator as recited in claims2-3, 7-8 and 11-12. Thus, the Office Action uses Denton to cure these deficiencies of Scott and Thompson and combines the teachings of Denton, Scott and Thompson to yield the claimed invention. The rejection is traversed as being based on references that neither teach nor suggest the novel combination of features clearly recited in independent claims 1, 6 and 10, upon which each of claims 2-3, 7-8 and 11-12 depend. Furthermore these dependent claims recite addition limitations which are not shown in the cited prior art.

Denton does not cure the deficiencies of Thompson and Scott. Denton teaches an optical networking module that is formed with an integrated module including optical, optical-electrical and protocol processing components and complementary software.

Each of claims 2-3, 7-8 and 11-12 depend on claims 1, 6 and 10 respectively, and thus, incorporates all of the elements of the independent claims. As such, each of claims 2-3, 7-8 and 11-12 include a header detector configured to detect a header cell of the data packet and remove a header from the header cell of the data packet; a counter to determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed; an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes, as recited in claims 1, 6 and 10. There is simply no teaching or suggestion in Denton of a header detector configured to detect a header cell of the data packet and remove a header from the header cell of the data packet; a counter to determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed; an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes, as recited in claims 2-3, 7-8 and 11-12. Therefore, Applicants respectfully assert that the rejection under 35 U.S.C. §103(a) should be withdrawn because neither Scott, Thompson, nor Denton, whether taken singly or combined, teaches or suggests each feature of claims 1, 6 and 10 and hence dependent claims 2-4, 7-8 and 11-12, thereon.

Claims 4-5, 9 and 13 were rejected under 35 U.S.C. 103(a) as being obvious over Thompson in view of Scott and further in view of U.S. Patent No. 6,122,279 to Milway and U.S. Patent No. 6,697,873 B1 to Yik. According to the Office Action, Thompson and Scott teach all of the elements of claims 1, 6 and 10 except for teaching a network element comprising a network switch as recited in claims 4-5, 9 and 13. Thus, the Office Action uses Milway and Yik to cure these deficiencies of Scott and Thompson and combines the teachings of Milway, Yik, Scott and Thompson to yield the claimed invention. The rejection is traversed as being based on references that neither teach nor suggest the novel combination of features clearly recited in independent claims 1, 6 and 10, upon which each of claims 4-5, 9 and 13 depend.

Milway also does not cure the deficiencies of Scott and Thompson. Milway teaches an ATM switch with a microprocessor, a switch controller, a memory, a token grant logic and port clusters, wherein each port cluster contains line interfaces, port logic and buffering for up to eight ATM network connections. Col. 6, lines 27-33. Cell data is delivered from one cluster to another by a switch bus. Col. 7, lines 3-4. The principal task of the switch is to route ATM cells from a plurality of input links to a plurality of output links. Col. 7, lines 22-24. In operation, ATM cells are received by the switch via an ATM line interface. A new ATM cell arriving on an input link is converted from electrical signals to a bit stream that is provided to a network control logic which checks for errors and discards misdirected cells. Col. 9, lines 1-11.

Yik also does not cure the deficiencies of Scott, Thompson and Milway. Yik teaches an apparatus and method for storing and searching computer node addresses in a computer network system. Each of claims 4-5, 9 and 13 depend on claims 1, 6 and 10 respectively, and thus, incorporates all of the elements of the independent claims. As such, each of claims 4-5, 9 and 13 include a header detector configured to detect a header cell of the data packet and remove a header from the header cell of the data packet; a counter to determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed; an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes, as recited in claims 1, 6 and 10.

There is simply no teaching or suggestion in Milway and Yik a header detector configured to detect a header cell of the data packet and remove a header from the header cell of the data packet; a counter to determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed; an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes, as recited in claims 1, 6 and 10. Therefore, Applicants respectfully assert that the rejection under 35 U.S.C. §103(a) should be withdrawn because neither

Scott, Thompson, Milway nor Yik, whether taken singly or combined, teaches or suggests each feature of claims 1, 6 and 10 and hence dependent claims 4-5, 9 and 13 thereon.

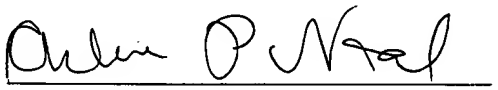
Furthermore, Applicants respectfully submit that the Office Action has pieced together four references to teach the claimed invention. However, MPEP 2143.01 instructs that "[t]he mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. In re Mills, 916 F.2d 680, 16 USPQ 2d 1430 (Fed. Cir. 1990)." MPEP 2143.01 further instructs that "[a]lthough a prior art device 'may be capable of being modified to run the way the apparatus is claimed, there must be a suggestion or motivation in the reference to do so.'" Applicants respectfully submit that the cited references do not provide such a suggestion or motivation. Applicants submit that the only motivation to piece together the four references of the Office Action is found in Applicants' own application. MPEP 2141, under the heading "Basic Consideration Which Apply to Obviousness Rejections," points out that "the references must be viewed without the benefit of impermissible hindsight vision afforded by the claimed invention." (See also Hodosh v. Block Drug Co., Inc. 786 F.2d 1136, 229 USPQ 182 (Fed. Cir. 1986).) The Federal Circuit has clearly held that "the motivation to combine references cannot come from the invention itself." Heidelberger Druckmaschinen AG v. Hantscho Commercial Products, Inc., 21 F.3d 1068, 30 USPQ 2d 1377 (Fed. Cir. 1993).

As noted previously, each of claims 1-13 recite subject matter which is neither disclosed nor suggested in the prior art references cited in the Office Action. It is therefore respectfully requested that all of claims 1-13 be allowed and this application passed to issue.

If for any reason the Examiner determines that the application is not now in condition for allowance, it is respectfully requested that the Examiner contact, by telephone, the applicant's undersigned attorney at the indicated telephone number to arrange for an interview to expedite the disposition of this application.

In the event this paper is not being timely filed, the applicant respectfully petitions for an appropriate extension of time. Any fees for such an extension together with any additional fees may be charged to Counsel's Deposit Account 50-2222.

Respectfully submitted,



Arlene P. Neal
Registration No. 43,828

Customer No. 32294
SQUIRE, SANDERS & DEMPSEY LLP
14TH Floor
8000 Towers Crescent Drive
Tysons Corner, Virginia 22182-2700
Telephone: 703-720-7800
Fax: 703-720-7802

APN:kzw